Timex Sinclair User Group NEWSLETTER

DEC 1987

Ottawa Chapter

EDITOR — M. Dove

Volume IV Issue III

TABLE OF CONTENTS

Meeting/Ad Rates 0001

From the Editor.......... 0010

Meetings in Review....... 0010

Copyright Information.... 0011

2068/ZX81 Programs....... 0111

ADVERTISING RATES

Members - 50 words or less.
(No charge)

Non-Members - 50 words or less
($5.00)

Commercial

Full page	*$30.00*
Half page	*$22.00*
Quarter page ..	*$12.00*
Eighth page ...	*$ 8.00*

MEETINGS

LOCATION: All meetings are held at the NRC bldg,. 100 Sussex Dr. Ottawa. Check with security when signing in for the room number.

TIME:Meetings are held at 8.00 P.M. the second Tuesday of every month (no meetings in July or August).

FORMAT:
1)-President's Address and introduction of new members.
2)-Software review (max of 20 minutes).
3)-Hardware review (max of 20 minutes).
4)-Adjourment of the formal part of the evening.
5)-Informal: Members discuss common problems, swap programs, and exchange info (tips) etc.

MEETING DATES

JAN	12th
FEB	9th
MAR	8th
APRIL	12th
MAY	10th
JUNE	14th
SEPTEMBER	13th
OCTOBER	11th
NOVEMBER	8th
DECEMBER	13th

MAILING ADDRESS

Please Address all correspondence to:

Michael Dove
The Editor
OHTSUG
2161 East Acres Rd.,
Gloucester, Ontario
Canada K1J 9A6

FROM: THE EDITOR

In case you do not have our numbers then read on.

President - Alan Moore
(226-7118)
Admin. - Michael Dove
(745-1779)
Software - Bill Harmer
(722-7184)
Hardware - John Mathewson
(746-7869)
Librarian - David Solly
(731-2120)
Treasurer - Iris Ten Holder
(828-6668)

MEETINGS IN REVIEW

For those of us who are not always able to attend our monthly meetings you might be interested in a summary of the highlights of those meetings (September through to November 1987).

At our September meeting, John Mathewson demonstrated his prototype speech synthesizer. At present it is capable of handling 64 basic (different) sounds. It was quite impressive and readily understandible. John hopes to have the bugs worked out in the next couple of months. He uses a ZX81 power supply to run the unit. Iris Ten Holder demonstrated her Payroll program. It is a ZX81 program which is used with a 64K Rampack. Iris uses this program to prepare weekly and monthly paystubs for her restaurant employees. Dave Solly demonstrated a modified program which utilized German and Hebrew character sets instead of the normal 2068 characters. He was then able to have it printed out on his printer.

Our November meeting was taken up almost entirely by discussions surrounding the possiblity of setting up our own bulletin board. John indicated that he could get hold of some non functional modem boards. He would be willing to try and repair them at cost for those who would like to have a modem. He indicated that he woudn't object to using one of his 2068 coupled to a disc drive to operate a bulletin board. Discusion on whether this would operate 24 hours of the day or at specific hours will be discussed later. The interest by members that presently had boards and those who would like to have modems makes it feasible to run our own system.

At our November meeting the repaired modems were distributed. Most of the meeting was spent familiarizing individuals with the pros and cons of software to be used with the modems. The modems and software would allow ZX81 and 2068 users to communicate directly with each other or via a bulletin board. The discussions on the bulletin board will continue at the December meeting. Clem Holden gave a demonstration on one of his current programs.

COPY RIGHT

The issue of copy right is a grey area and not understood by most of us. Following are exerpts of information on copy right coming from Bill Harmer, the DALLAS TIMEX/SINCLAIR AMSTRAD USER NEWSLETTER and the July/August 1987 issue of Canada Computes.

Of increasing concern to computer users and amateur programmers is the matter of copyright protection of our own and others' programs. Unfortunately there is more misinformation circulating on this subject than real knowledge. So it was with great relish that the writer viewed the opportunity to learn more about this subject from a lecture by a young lawyer from the local firm of Gowling and Henderson at a meeting in May of the Ottawa IBM PC User Group.

Please bear in mind that what follows is a combination of reportage and the attempts of the author to explain some of the concepts that were brought up at the lecture and must not be construed as absolutely certain legal advice for which (if it exists even in theory) you must consult a copyright lawyer yourself.

One of the things that it is possibly easiest to explain are some of the misconceptions that the writer has heard over the grapevine.

Misconception: when a program listing is published in a newsletter, magazine or book some of or all of the restrictions on use of the program are forfeited by the copyright holder. Some even have said that it thus becomes public domain.

According to the lecture nothing could be further from the truth. Not only are none of the protective or restrictive features of a copyrighted program lessened by publication but in fact the copyright owner's position is strengthened legally since the appearance of the book or magazine on a specific date is additional proof if needed of date of publication, a part of the copyright's legal support.

So any program published in listing form in fact cannot necessarily be stored in your computer (that would be a copy), copied on tape or disk, or the copies given to anyone else, or the program republished under your own credit or copyright on the basis that you had made some modifications or improvements to it.

The question of what you can do with a a published program depends entirely on the policy of the owner of the copyright and whether he/she has ever granted any of the above rights to readers of the magazine for example. This policy may be stated somewhere in the magazine or book usually beside the copyright and publisher's name or it may not be anywhere and you may have to write the publisher to find out what the policy is.

Misconception: to copyright one of your own original programs, you must apply to the government, submit a copy of the program and pay a fee. Some even go as far as to say that you must hire a lawyer to get a copyright.

Again, nothing is further from the truth. Any program can be copyrighted and in fact from the time of its creation enjoys copyright protection in the name of the author by simply publishing it with a notice such as, "Copyright 1987 by John Doe". All the rest including registering the copyright with the federal Department of Consumer and Corporate Affairs (for a fee or $35, no copy need be submitted), hiring a lawyer for legal advice and to dot all the "i"s and cross all the "t"s and take care of the paperwork is intended only to consolidate the legal position of the copyright to the point where it would be harder for any other person to challenge it on a point of legal detail or find some defect or loophole or way to circumvent the copyright holder's protection on a technicality. How much trouble and expense you want to go to to make your copyright ironclad depends as with insurance coverage limits on how valuable you think the item (in this case the program) is that you are trying to protect. If you think that the program will make you wealthy hire a lawyer.

If you think that the program is just another of many that you have written with little chance of making any money for you it wouldn't make all that much sense to run up a big bill at the lawyer's office for it now would it?

Although there are a number of ways of protecting computer programs in special circumstances including patents, trade secret laws and trade mark laws, the most commonly used at our level of sophistication are copyrights and contracts.

Copyrights can be used in combination with contracts to either give away some of the rights of copyright or to add to them. Armed with a copyright you can try some way of getting the user to sign or in some other way (a common way is putting a notice inside the shrinkwrap package around a program saying that if you break open the shrinkwrapping you are agreeing to the often many terms of a contract between you and the manufacturer/distributor restricting your use of the program) enter into a contract with the copyright holder adding additional restrictions and terms of use to those of the copyright. These restrictions might include such things as absolving the manufacturer for liability for program bugs or damage to equipment due to running it, restricting your right to make copies to a given number (say one back-up copy), restricting your right to use the program in more than one computer etc.

On the other hand the author can give away any or all of his copyright protection or aspects of it. He/she might on the package or within the program grant free rights of use to members of the local user group, inhabitants of third world countries, teachers or educators, or whatever. In so called freeware, instead of issuing the program as public domain and including in it a solitictation for funds such as, "If you enjoyed this program, please send $20 to the author at such and such an address", you can protect some of your rights (commercial firms might remove your solicitation from the program and sell it or bundle it with other products) by copyrighting the program and then granting certain rights to the general public. An example would be a notice such as, "Feel free to make copies of this program for your own personal use and for the inspection of your friends and if you enjoyed this program and wish to receive any up-dates or modifications and a nice user manual for it, please send $25 to the author...." Most of the so-called freeware etc. is now marketed in that way rather than casting it forth upon the waters in the public domain.

In addition to legal rights of the programmer/author, the legal obligations were also discussed. For example a programmer could be prosecuted for negligence if he/she didn't take proper time to test it for bugs and one of the bugs caused a user to take a financial loss. Also copy-protection methods that trash the disk and the data entered even by an illegal user could be the subject of criminal code prosecution of the programmer under the Mischief section of the Criminal Code of Canada. Also the Sale of Goods Act and various pieces of consumer legislation might be invoked for faulty software. We all know that you can't write software without bugs, but the law demands that you make a conscientious effort to find them before the program goes out. On the other hand for the sort of hobby programs that most of us write the financial losses due to a bug are unlikely to be worth suing us for. Who is going to sue us for an hour of wasted time or data entry or a frustrating game?

All and all, the lecture was a very interesting one , and of course lasting an hour and a half covered much more than reported above.

EVERYTHING YOU EVER WANTED TO XEROX ABOUT COPYRIGHTS.

-(C) 1987 in the Public Domain-

I want to explain a little I have found out about copyright law, because it is really getting to be an important subject for everybody in the computer world, not just for artists and playwrights and novelists. Now it is necessary for Programmers to know what part of their work, if any, is protected from use or reproduction without their consent.

It will anger some to hear the truth being spoken. You bought a program (or record or tape or whatever) that was a commercial product, yours to use. The copyright prohibits you from making copies to sell or give away. It does not say you can't make copies for your own private use, once you bought the original product. Copyright does not say backups are against the law. Copyright does not say that you as the buyer are not entitled to choose when or where you use the product you bought, once again, as long as you are not making it available to a non-private use. In fact, the only thing that says these things are wrong is the software license that the lawyers of the software company drafted in order to intimidate a potential copyright abuser. The License is a pre-sale contract with the user. Read it. Don't buy software from stupid companies. Keep in mind that breach of this contract is a civil matter; they may choose to sue you over it. Copyright violations are not the same thing, not at all. Copyright violation is a Felony offense.

Backups and additional private use copies are allowed under the copyright laws due to a section in them called FAIR USES. Fair Use was written so that people could properly deal with a work, such as in reviewing or using it for private study. Private use means that if the only market you are spoiling through your actions is YOU, and you already bought the work once, no copyright violation exists. Additionally, Fair use means you in no way are competing with the work itself and what you are doing with the work is for educational, non-commercial purposes, such as extracting portions of a work for reporting on it or the events which surround it.

You are NOT allowed to "spoil a market" for someone else. Don't give away copies. Don't sell any copy (including the original) and keep any part of the work. Don't republish the work or broadcast it in any fashion. Co-workers and friends who want a copy should buy it themselves. But if you abide by these things, the government doesn't care if you personally have made 3000 duplicates for Yourself (one for each of your phobias?). The software companies see this as practically an INVITATION to pirate, so they don't want you aware of the full law. Guilty until proven innocent... that is the motto of some of the companies.

So now that you know... what about compilers? They help you create your programs, but they do this by stringing together little stored pieces of code that they wrote. Can you then sell your stand-alone applications program? Not necessarily. The little pieces are integral to the operation of the compiler and they are a part of the compiler's copyright. They are mostly unique to the compiler. Now they are a part of your program, too, which is O.K. since you can use the program yourself; but if you are creating a program to sell, you had better darn sure check into the OBJECT CODE DISTRIBUTION requirements the compiler's creator specifies. Some companies demand royalties on any program developed with their product. My favorite compiler is MTBASIC, by Softaid, Inc. Their policy is that you can develop anything you want to and sell it, providing that in the documentation and code of a program you have a credit, such as: "Compiled using MTBASIC, Softaid Inc., P.O. Box 2412, Columbia MD 21045". I don't have any objection to plugging a fine tool I like, so this object code requirement is the kind of thing I wish every compiler maker would take to heart.

What about magazine listings? In the Home Computer world, this is a major source of programs. Need I tell you that most all of them are copyrighted? They were virtually all written by users like you and me, who wished to share them with the community of micro users that could be reached through the magazine. Public Domain style. Then the magazine paid them a five-dollar fee, had them sign a form giving them exclusivity, and published it under the magazine's copyright. The magazine usually has a blurb on the contents page where you would never read it, which states, "Material contained herein is the property of HappyHome Computing, Inc. and may not be reproduced in any form without obtaining express written permission from the publisher." Great. Instead of Public Domain, which can go onto a Bulletin-Board, for instance, we have Fair Use Copyright, which can be typed into your computer but not shared. My feelings on this are that it is not fair, and that it can be licked. The program itself is showing you a Source Algorithm, a solution to a particular problem done up in a language you use on your computer. Now pick up the ball... and rewrite the program completely anew. Sure it is tougher, but the result will be Public Domain. Adapt Commodore listings to your Timex. The resulting code IS NEW CODE. No doubt about it. Your programming skills will be better, too. Change stupid input chains to menus... write a different sort routine. Listings are to learn by, anyway. If all you want is something for yourself, it doesn't matter. If you want to share, contribute ORIGINALLY.

And that's what all this copyright stuff is about. It is simply a matter of keeping the lazy and the dishonest from ruining someone's chances of doing the work and doing it well enough to earn something, even if it's just recognition, as is mostly the case in home computing. And when you write a program, consider the possibility of sharing it with others in the Public Domain, which builds a base upon which education and enlightenment for your computer can grow. Healthy computer communities need people and their programs to show the way. If you copyright, you are part of a market that is regulated and pretty secure-- now all you need is that skill and idea that will set the computer world on its' ear. Good Luck!

<< THIS ARTICLE IS DECLARED SEPARATE FROM THE REST OF THE DALLAS TIMEX/SINCLAIR AMSTRAD USERS NEWSLETTER, FOR THE PURPOSE OF DEDICATING IT TO THE PUBLIC DOMAIN. THE AUTHOR'S INTENT IN DOING THIS IS TO INSURE THE EDUCATION OF COMPUTERISTS EVERYWHERE. >> - and I wanted to make sure there was at least One example of P.D. available....

################################# See you at the meeting!

New law aims to put pirates out of business

By SHARON CHALUPIAK

The Copyright Act will state clearly it is illegal to copy computer programs, if the federal government passes the proposed amendments introduced into the House of Commons on May 27.

Computer programs will be defined as literary works and have the same copyright protection as books and songs—the lifetime of the author and 50 years thereafter.

Until now software publishers have been trying to use the existing act to prosecute anyone copying their programs.

But it is difficult to prosecute under the Copyright Act because it does not mention computer programs specifically as being protected. The act was enacted in 1924, before computers were invented.

Software publishers are uncertain where they stand.

The Federal Court of Canada did decide in one case that computer software is creative work in either written form or machine language and is therefore entitled to copyright protection. But the decision is being appealed.

In a precedent-setting criminal case, a Toronto software copier was sentenced to five months in jail after being found guilty of fraud for copying software programs. But that case is also being appealed.

Even if publishers succeed in prosecuting a business under the act, the maximum penalty is a $200 fine, which is not much of a deterrent to businesses capable of making much more from pirated software.

One in four programs on the market is estimated to be pirated, causing the industry to lose millions of dollars annually to unauthorized copying.

The amended Copyright Act would refer specifically to computer programs.

Penalties will increase to a maximum of five years in jail or a $1 million fine.

Program manufacturers say they are reluctant to develop new programs if they will lose profits because the programs can be easily copied without penalty. Software copying businesses say they increase the popularity of original programs and actually benefit program manufacturers.

However, one man whose business has been built around copying programs says he may favour the proposed legislation.

"If it brings the price of software down, it's an excellent law," says James Leahy, owner of Clubsoft, a software evaluation club in Toronto.

Prices should drop because program manufacturers have claimed price has to cover lost sales due to piracy, he says.

But consumers could be hurt if publishers use the copyright protection to raise prices, says Leahy.

The legislation will result in other amendments including changing the royalty rate to songwriters.

Further amendments are to be introduced in the fall.

No charges after Clubsoft raid

Clubsoft has yet to be charged two months after an RCMP raid in which police loaded two vehicles with Clubsoft hardware and programs.

"Business is booming," says James Leahy, owner of the Toronto business which calls itself a software evaluation club.

Before the raid the company, calling itself a software evaluation club, was making copies of original programs at low prices for members to evaluate to see if they wanted to buy the original programs, according to manager James Leahy.

Clubsoft is still carrying on the business of selling original software. It also continues to rent original software to its members. If the members don't like the program

FOR SALE

BROTHER HR-5 THERMAL DOT MATRIX PRINTER

INTERFACE: Parallel Centronics or RS-232
PRINTING SPEED: 30 cps
NO. COLUMNS: 80-90, 40-48, 66, 132
PAPER: single or continuous. Standard paper or thermal paper.
PRICE: $150.00

Andre Laviolette
1385 ouest, av. Bernard suite 16
Montreal QUEBEC H2V 1W1
Phone: (514) 273 4103

HINTS FOR THE 2068

The following information came from "TS HORIZON" (JULY 1986). You may not have seen this article written by George Mockridge. So I am presenting it for your information. There are some really good tips here.

1. POKE 23609,X
   For keyboard click (1-255)
2. POKE 23692,2
   Scroll (can use USR 3582)
3. POKE 23692,1
   Scroll 22 lines
4. POKE 23658,8
   Engage caps mode
5. POKE 23658,0
   Disengage caps mode
6. PAUSE 0
   Pause till any key pressed
7. POKE 23561,N (N = 1 TO 35)
   Interval of repeat key
8. POKE 23562,N (N = 1 TO 5)
   Delay interval in repeat key
9. USR 15002
   Break out of an infinite input loop w/o crashing
10. DIM A$ (704)
    PRINT AT 0,0; OVER 1; PAPER 1; INK6; A$
    Allows change to paper and ink color w/o CLS
11. PRINT #1; AT 0,2; "HI"
    PRINT #1; AT 1,5; "BY"
    PAUSE 0
    Prints on line 22/23
12. LOAD ""CODE
    RAND USR 33792
    For programs which will not load
13. RAND USR 0
    Will reset computer
14. INPUT AT 22,0; AT 10,0; "INPUT VALUE";A$
    Input at any position on screen
15. OPEN # 2
    Sends all data normally destined for the screen to the printer
16. CLOSE # 2
    Cancels above command
17. BASIC STARTS AT 26710
18. CAPS SHIFT 3
    Scrolls two screen when listing
19. POKE 23659,0
    To use all 24 lines (unable stop program) POKE 23659,2 resets. Use with INKEYS only
20. POKE 26710,255
    Use to make lines disappear. POKE 26710,0 will reset
21. POKE 23617,236
    Use ti get a question mark in input statement
22. PRINT # 0;"COMMENT"; PAUSE 0. Use to print on line 24.

CK TYPE

A NEW STANDARD FOR "FOOL-PROOF" TYPING OF TS2068 PROGRAM LISTINGS IN MAGASINES.

This program was originally published in the Mar/Apr 1987 issue of TIME DESIGNS. I was very impressed by this simple yet elegant program. It is also very useful. How many people have typed in large programs and then spent hours trying to find and correct errors. This program gives a line by line listing of the number of bytes used and a syntatic sum for each line. Run this program with your typed in version, compare it to the published listing and it should be easy to locate your errors. I have taken the liberty of converting this program to the ZX81.

```
9990 REM  this basic program
was taken from STAN LEMKE list_
ing, TIME DESIGNS (MAR/APR 87)
9991 LET start=256*PEEK 23636+PE
EK 23635: LET stop=256*PEEK 2362
8+PEEK 23627: LET byt=start
9992 LET line=0: LET lenght=0: L
ET sum=0
9993 LET line=256*PEEK byt+PEEK
(byt+1): LET byt=byt+2
9994 LET lenght=256*PEEK (byt+1)
+PEEK byt: LET byt=byt+2
9995 LET odd=1: FOR i=1 TO lengh
t: LET sum=sum+(odd+1)*PEEK byt:
 LET odd=NOT odd: LET byt=byt+1:
 NEXT i
9996 PRINT line;TAB 6;": ";lengh
t;TAB 13;": ";sum
9998 IF byt<stop THEN GO TO 9992
9999 STOP
```

EXAMPLE

```
9990  : 87  : 10530
9991  : 96  : 8966
9992  : 43  : 4178
9993  : 53  : 5151
9994  : 55  : 5581
9995  : 89  : 10292
9996  : 46  : 4490
9998  : 22  : 2843
9999  : 2   : 465
```

```
9980 REM THIS BASIC PROGRAM
WAS TAKEN FROM STAN LEMKE LIST-
ING, TIME DESIGNS (MAR/APR 87)
9981 LET START=16509
9982 LET STOP=256*PEEK 16401+PEE
K 16400
9983 LET BYT=START
9984 LET LINE=0
9985 LET LENGHT=0
9986 LET SUM=0
9987 LET LINE=256*PEEK BYT+PEEK
(BYT+1)
9988 LET BYT=BYT+2
9989 LET LENGHT=256*PEEK (BYT+1)
+PEEK BYT
9990 LET BYT=BYT+2
9991 LET ODD=1
9992 FOR I=1 TO LENGHT
9993 LET SUM=SUM+(ODD+1)*PEEK BY
T
9994 LET ODD=NOT ODD
9995 LET BYT=BYT+1
9996>NEXT I
9997 PRINT LINE;TAB 6;": ";LENGH
T;TAB 13;": ";SUM
9998 IF BYT<STOP THEN GO TO 9984
9999 STOP
```

OTHELLO

This program was given to me several years ago by Jim Webb (a former ZX81 user). He had seen this program listed in a magazine for the Commodore 64. This program as listed wil run on both the ZX81 and the TS2068. The only change would be at line 9998 which must be rewritten to make it auto run for the specific machine. It is a good program and enjoyable.

```
 120 DIM A(9,9)
 121 DIM G(8)
 122 DIM H(8)
 123 DIM C$(8)
 124 DIM D$(3)
 140 PRINT "GREETINGS FROM OTHEL
LO"
 141 PRINT
 142 PRINT "MODIFIED FOR ZX81 BY
 **JIM WEBB,DEC.82**"
 143 PRINT
 144 PRINT
 150 PRINT "INSTRUCTIONS (Y OR N
)"
 160 INPUT X$
 170 IF X$="N" THEN GO TO 390
 180 IF X$<>"Y" THEN GO TO 160
 190 PRINT "**"
 200 CLS
 201 PRINT "OTHELLO IS PLAYED ON
 AN 8X8      CHECKER BOARD,"
 210 PRINT "ROWS:1-8; COLUMNS:A-
H"
 220 PRINT "INITIAL BOARD IS ALL
 BLANK,      EXCEPT FOR THE CENTR
E 4 SQUARES"
 230 PRINT "THESE FORM THE PATTE
RN"
 240 PRINT "       * O"
 250 PRINT
 255 PRINT "       O *"
 270 PRINT
 273 PRINT "ENTER SPACE TO CONTI
NUE"
 274 INPUT Z$
 275 IF Z$<>" " THEN GO TO 274
 280 CLS
 290 PRINT "TRY TO PLACE YOUR PI
ECE SO THAT IT OUTFLANKS MINE,CR
EATING A ";
 295 PRINT "HORIZONTAL,VERTICAL
OR DIAGONAL RUN OF MY PIECES BOU
NDED AT EACH"
 300 PRINT " END BY AT LEAST ONE
 OF YOURS."
 301 PRINT
 302 PRINT "ENTER SPACE TO CONTI
NUE"
 303 INPUT Z$
 304 IF Z$<>" " THEN GO TO 303
 305 CLS
 310 PRINT "THIS WILL FLIP MY PI
ECES,TURNING THEM INTO YOURS."
 320 PRINT "NOTE***,YOU MUST CAP
TURE AT      LEAST ONE OF MINE IN
 THIS WAY IF IT IS POSSIBLE "
 330 PRINT "OTHERWISE YOU FORFEI
T A TURN BY ENTERING (0,0)"
 380 PRINT "ENTER SPACE TO CONTI
NUE"
 381 INPUT Z$
 382 IF Z$<>" " THEN GO TO 381
 383 CLS
 390 PRINT "SHOULD I WAIT?(Y OR
N)"
 400 PRINT
 405 LET F2=0
 410 INPUT X$
 420 IF X$="N" THEN GO TO 460
 430 IF X$<>"Y" THEN GO TO 410
 440 LET F2=1
 450 PRINT "TYPING ANY CHARACTER
 LETS ME GO"
 460 PRINT "SHOULD I USE MY BEST
 STRATEGY?(Y OR N)"
 470 LET S2=0
 480 INPUT X$
 490 IF X$="N" THEN GO TO 520
 500 IF X$<>"Y" THEN GO TO 480
 510 LET S2=2
 520 LET B=-1
 530 LET W=1
 540 LET D$(B+2)="O"
 550 LET D$(0+2)="."
 560 LET D$(W+2)="*"
 561 LET G(1)=0
 562 LET G(2)=-1
 563 LET G(3)=-1
 564 LET G(4)=-1
 565 LET G(5)=0
 566 LET G(6)=1
 567 LET G(7)=1
 568 LET G(8)=1
 620 LET H(1)=1
 621 LET H(2)=1
 622 LET H(3)=0
 623 LET H(4)=-1
 624 LET H(5)=-1
 625 LET H(6)=-1
 626 LET H(7)=0
 627 LET H(8)=1
 660 LET C$(1)="A"
 661 LET C$(2)="B"
 662 LET C$(3)="C"
 663 LET C$(4)="D"
 664 LET C$(5)="E"
 665 LET C$(6)="F"
 666 LET C$(7)="G"
 667 LET C$(8)="H"
 700 FOR I=1 TO 9
 710 FOR J=1 TO 9
 720 LET A(I,J)=0
 730 NEXT J
 740 NEXT I
 750 LET A(4,4)=W
 760 LET A(5,5)=W
 770 LET A(4,5)=B
 780 LET A(5,4)=B
 790 LET C1=2
 800 LET H1=2
 810 LET N1=4
 820 LET Z=0
 840 PRINT "DO YOU WANT O OR * ?
"
 850 LET C=W
 860 LET H=B
 870 INPUT X$
 880 IF X$="O" THEN GO TO 920
 890 IF X$<>"*" THEN GO TO 870
 900 LET C=B
 910 LET H=W
 920 PRINT "DO YOU WANT TO GO FI
RST(Y OR N)"
 930 INPUT X$
 940 IF X$="N" THEN GO TO 1020
 950 IF X$<>"Y" THEN GO TO 930
 965 CLS
 970 GO SUB 3100
 980 GO TO 1690
1000 IF F2=0 THEN GO TO 1020
1010 INPUT X$
1020 LET B1=-1
1030 LET I3=0
1040 LET T1=C
1050 LET T2=H
1070 FOR I=1 TO 8
1080 FOR J=1 TO 8
1090 IF A(I,J)<>0 THEN GO TO 138
0
1120 GO SUB 2620
1130 IF F1=0 THEN GO TO 1380
1170 LET U=-1
1180 GO SUB 2820
1200 IF S1=0 THEN GO TO 1380
1210 IF (I-1)*(I-8)<>0 THEN GO T
O 1230
1220 LET S1=S1+S2
1230 IF (J-1)*(J-8)<>0 THEN GO T
O 1260
```

```
1240 LET S1=S1+S2
1260 IF S1<B1 THEN GO TO 1380
1270 IF S1>B1 THEN GO TO 1340
1310 LET R=RND
1320 IF R>0.5 THEN GO TO 1380
1340 LET B1=S1
1350 LET I3=I
1360 LET J3=J
1380 NEXT J
1390 NEXT I
1410 IF B1>0 THEN GO TO 1480
1430 PRINT "I HAVE TO FORFEIT MY
 MOVE"
1440 IF Z=1 THEN GO TO 2190
1450 LET Z=1
1460 GO TO 1690
1480 LET Z=0
1490 PRINT "I WILL MOVE TO ";
1500 PRINT I3;
1510 PRINT " ";
1520 PRINT C$(J3)
1530 LET I=I3
1540 LET J=J3
1550 LET U=1
1560 GO SUB 2820
1570 LET C1=C1+S1+1
1580 LET H1=H1-S1
1590 LET N1=N1+1
1595 CLS
1600 PRINT "THAT GIVES ME ";
1610 PRINT S1;
1620 PRINT " OF YOUR PIECES"
1640 GO SUB 3100
1660 IF H1=0 THEN GO TO 2190
1670 IF N1=64 THEN GO TO 2190
1690 LET T1=H
1700 LET T2=C
1710 PRINT "YOUR MOVE-- (ROW,COL
UMN)";
1720 INPUT I
1721 INPUT X$
1730 IF I<0 THEN GO TO 1720
1740 IF I>8 THEN GO TO 1720
1750 IF I<>0 THEN GO TO 1820
1760 PRINT "ARE YOU FORFEITING Y
OUR MOVE?";
1770 INPUT X$
1780 IF X$<>"Y" THEN GO TO 1710
1790 IF Z=1 THEN GO TO 2190
1800 LET Z=1
1810 GO TO 1000
1820 LET J=0
1825 FOR P=1 TO 8
1826 LET J9=P
1830 IF C$(J9)=X$ THEN GO TO 183
5
1832 GO TO 1840
1835 LET J=J9
1836 LET J9=8
1840 NEXT P
```

```
1850 IF J=0 THEN GO TO 1720
1870 IF A(I,J)=0 THEN GO TO 1910
1880 PRINT "SORRY,OCCUPIED: TRY
AGAIN"
1890 GO TO 1720
1910 GO SUB 2620
1920 IF F1=1 THEN GO TO 1970
1930 PRINT "SORRY,YOU ARE NOT NE
XT TO ONE OF MY PIECES,TRY AGAIN
"
1950 GO TO 1720
1970 LET U=-1
1980 GO SUB 2820
1990 IF S1>0 THEN GO TO 2030
2000 PRINT "SORRY,THAT DOESNT FL
ANK A ROW.TRY AGAIN"
2010 GO TO 1720
2030 LET Z=0
2035 CLS
2040 PRINT "THAT GIVES YOU ";
2050 PRINT S1;
2060 PRINT " OF MY PIECES"
2070 LET U=1
2080 GO SUB 2820
2090 LET H1=H1+S1+1
2100 LET C1=C1-S1
2110 LET N1=N1+1
2130 GO SUB 3100
2150 IF C1=0 THEN GO TO 2190
2160 IF N1=64 THEN GO TO 2190
2165 PRINT "PRESS SPACE FOR ME T
O GO";
2166 INPUT N$
2167 IF N$<>" " THEN GO TO 2166
2170 GO TO 1000
2190 PRINT
2200 PRINT "YOU HAVE ";
2210 PRINT H1;
2220 PRINT "PIECES # I HAVE ";
2230 PRINT C1;
2240 PRINT "PIECES"
2250 IF H1=C1 THEN GO TO 2290
2260 IF H1>C1 THEN GO TO 2310
2270 PRINT "SORRY,I WON THAT ONE
***"
2280 GO TO 2320
2290 PRINT "A TIE*"
2300 GO TO 2500
2310 PRINT "YOU WON**"
2320 LET C1=C1-H1
2330 IF C1>0 THEN GO TO 2350
2340 LET C1=-C1
2350 LET C1=(64*C1)/N1
2360 PRINT "THAT WAS A ";
2370 IF C1<11 THEN GO TO 2490
2380 IF C1<25 THEN GO TO 2470
2390 IF C1<39 THEN GO TO 2450
2400 IF C1<53 THEN GO TO 2430
2410 PRINT "**PERFECT GAME**"
2420 GO TO 2500
2430 PRINT "WALKAWAY*"
```

```
2440 GO TO 2500
2450 PRINT "GOOD FIGHT*"
2460 GO TO 2500
2470 PRINT "HOT GAME**"
2480 GO TO 2500
2490 PRINT "SQUEAKER*"
2500 PRINT
2510 PRINT "ANOTHER GAME? (Y OR
N)"
2520 INPUT X$
2530 IF X$="Y" THEN GO TO 700
2540 IF X$<>"N" THEN GO TO 2520
2550 PRINT "THANKS FOR PLAYING"
2560 STOP
2620 LET F1=0
2630 FOR P=-1 TO 1
2631 IF I+P=0 THEN GO TO 2670
2640 FOR Q=-1 TO 1
2641 IF J+Q=0 THEN GO TO 2660
2650 IF A(I+P,J+Q)=T2 THEN GO TO
 2655
2651 GO TO 2660
2655 LET F1=1
2656 LET J1=1
2657 LET I1=1
2660 NEXT Q
2670 NEXT P
2680 RETURN
2820 LET S1=0
2830 FOR K=1 TO 8
2840 LET I5=G(K)
2850 LET J5=H(K)
2860 LET I6=I+I5
2865 IF I6=0 THEN GO TO 3070
2870 LET J6=J+J5
2875 IF J6=0 THEN GO TO 3070
2880 LET S3=0
2890 IF A(I6,J6)<>T2 THEN GO TO
3070
2910 LET S3=S3+1
2920 LET I6=I6+I5
2925 IF I6=0 THEN GO TO 3070
2930 LET J6=J6+J5
2935 IF J6=0 THEN GO TO 3070
2940 IF A(I6,J6)=T1 THEN GO TO 2
970
2950 IF A(I6,J6)=0 THEN GO TO 30
70
2960 GO TO 2910
2970 LET S1=S1+S3
2980 IF U<>1 THEN GO TO 3070
3000 LET I6=I
3010 LET J6=J
3020 FOR R=0 TO S3
3025 LET K1=R
3026 IF J6=0 THEN GO TO 3060
3027 IF I6=0 THEN GO TO 3060
3030 LET A(I6,J6)=T1
3040 LET I6=I6+I5
3050 LET J6=J6+J5
3060 NEXT R
```

```
3070 NEXT K
3080 RETURN
3100 PRINT
3110 PRINT "   A B C D E F G H"
3120 FOR I=1 TO 8
3130 PRINT I;
3140 FOR J=1 TO 8
3150 PRINT " ";
3160 PRINT D$(A(I,J)+2);
3170 NEXT J
3180 PRINT
3185>PRINT
3190 NEXT I
3210 RETURN
3220 STOP
9998 SAVE "OTHELLO"
```

The following programs have been given to me by Dave Solly. His documentation is as usual excellent. Read on.

Program Reader
for the
Timex-Sinclair 1000/1500

Have you at one time or another forgotten the name of a praticular Timxe-Sinclair 1000 program but didn't want to wait the 10 minutes and more to see what it was? Help is at hand! This routine will read and print the name the program was saved under, all in a few seconds and without affecting the saved program.

First, type in Listing 1 remembering the 75 character REM statement. This listing creates the machine code workhorse of the program. Save then run the program. Test the program by rewinding to the beginning of the tape and typing RAND USR 16514 and playing the tape. If the computer returns you to BASIC with the name of the program displayed on the screen then the machine code is ok. If the program does not return you to BASIC or if it crashes then there is most likely a mistake in the machine code. Reload the tape and check both the length of your REM statement and the data contained in A$ in line 10. Correct any mistakes, save, run and test the program again. If the program then runs without any problems you can proceed. I would avise at this point; however, that you keep at least one good copy of the first program.

Listing 1
Machine Code
and
Hex Loader

```
   1 REM [Note: 75 characters
must be typed in this REM
statement to hold machine code]
  10 LET
A$="CD230FCD8A4018FB0E0106003E7F
DBFED3FF1FD2A20317173811 10F1F1CD
8A40CB7A792001D71730F4181DD51E94
061A1DDBFE17CB7B7B38F510F5D12004
FE5630C83FCB1130C3C9C9S"
  20 LET X=16514
  30 IF A$="S" THEN STOP
  40 POKE X,16*CODE A$(1)+CODE
A$(2)-476
  50 LET X=X+1
  55 PRINT A$( TO 2);" ";
  60 LET A$=A$(3 TO )
  70 GO TO 30
  80 STOP
9000 REM SAVE ROUTINE
9005 PAUSE 100
9010 POKE 16390,9
9020 SAVE "READER.B1"
9040 IF PEEK 16390=9 THEN SAVE
"READER.B1"
9050 POKE 16390,0
9060 LIST
```

For those who wish you can delete lines 10 to the end, resave and use the program as it is. The machine code is relocateable and can be kept above ramtop for those of you who have larger memories or stored in a Hunter Board NVM. For my own use I have created Listing 2 which allows you to list several programs in succession on the screen without having to repeat typing the RAND USR call. Some of you might even want to add a screen copy routine in order to create a permanent record of the programs on a given tape.

Listing 2

Main Program with Instructions

```
   1 REM [Note: This will be the
75 bytes of machine code saved
in the REM statement from
Listing 1]
  10 REM LISTING 2 BY DAVID
SOLLY
  20 CLS
  30 PRINT "THIS PROGRAM WILL
ALLOW YOU TO  READ THE NAME
UNDER WHICH A      PROGRAM HAS
```

```
BEEN SAVED WITHOUT  HAVING TO
LOAD THE ENTIRE PRO-  GRAM."
  40 PRINT "THIS MAY BE USEFUL
IF THE      PROGRAM NAME HAS
BEEN FORGOTTEN OR IF THE PROGRAM
FAILS TO LOAD WITH: """"."
  50 PRINT
  60 PRINT "instructions: "
[Note: Typed in inverse
characters]
  70 PRINT
  80 PRINT "START THE TAPE TO BE
READ AND   HIT ANY KEY. THE
COMPUTER WILL  APPEAR TO BE
DOING A NORMAL     LOAD. WHEN A
""HEADER"" HAS BEEN  DETECTED
THE COMPUTER WILL      STOP AND
DISPLAY THE TAPE PRO-  GRAM
NAME."
  90 PRINT "HIT ""Y"" TO READ
ANOTHER HEADER  OR ANY OTHER KEY
TO STOP."
 100 IF INKEY$="" THEN GOTO 100
 110 CLS
 120 RAND USR 16514
 130 PRINT
 140 PRINT
 145 IF PEEK 16442<4 THEN
SCROLL
 150 PRINT "HIT ""Y"" TO READ
ANOTHER HEADER"
 155 PRINT
 160 IF INKEY$="" THEN GOTO 160
 170 IF INKEY$="Y" THEN RUN 120
 180 PRINT
 190 PRINT "TYPE ""RUN"" TO
RESTART PROGRAM."
8999 STOP
9000 REM SAVE ROUTNE
9005 PAUSE 100
9010 POKE 16390,9
9020 SAVE "READER.B3"
9040 IF PEEK 16390=9 THEN SAVE
"READER.B3"
9050 POKE 16390,0
9060 RUN
```

Program taken in part form the August 1986 edition of SMUG BYTES

1110

EYESIGHT SAVER

The following is a copy of an article which first appeared in Practical Computing, May 1984. Although the program was designed to run on the Specturm the following listing has been modified so that it will run on the Timex-Sinclair 2068.

Anyone who has copied programs from this or any other page will know it plays havoc with the eyesight if the coding is packed to maximum density. This is fine for memory economy but terrible for reading and copying. But hope is at hand. CP Marriner of Grinsby has developed a small routine which will shake up a program of multi-statement lines so that each statement is on a new line.

The Spectrum stores progams in memory from location 23755 or in the case of the Timex-Sinclair 2068 from location 26710. Each line of the program is then stored with two bytes for the line number and two bytes holding the length of the text, the text of the line and an end-of-the-line marker CHR$(13).

The routine lists any Basic program with the following changes. Every time a colon or the keyword Then is found a new line is started with the correct indentation to match the proceeding line. The result is a program listing which is not only easier to read but also a listing which is rendered in a format which much resembles structured Basic.

In practice the routine is merged with the programand suitably isolated. To print the formatted listing enter Goto 9000 or its equivalent if you have renumbered.

Program Listing

```
9900 REM Format listing
        TS 2068 Version
9905 OPEN#2,"p": LET dsp=0: REM Printer, spacing
9910 LET n=26710: REM Spectrum 23755
9915 IF dsp=1 THEN PRINT
9920 PRINT ': PRINT 256*PEEK n+PEEK (n+1);
9925 LET space=LEN STR$ (256*PEEK n+PEEK (n+1))+1
9930 IF 256*PEEK n+PEEK (n+1)>=9899 THEN OPEN #2,"s": PRINT
" O.K.": STOP
9935 FOR i=n+4 TO n+3+PEEK (n+2)+256*PEEK (n+3)
9940 IF PEEK i=58 THEN PRINT ":";TAB space;: GO TO 9960
9945 IF PEEK i>=32 THEN PRINT CHR$ PEEK i;
9950 IF PEEK i=14 THEN LET i=i+5
9955 IF PEEK i=203 THEN PRINT 'TAB space;
9960 NEXT i
```

```
9965 LET n=(n+3+PEEK (n+2)+256*PEEK (n+3)+1)
9970 GO TO 9915
9975 STOP
9980 SAVE "Llist": SAVE "Llist"
9985 SAVE "Llist": SAVE "Llist"
9986 CLS : PRINT "Verification: Rewind tape and   play."
9990 VERIFY "Llist": VERIFY "Llist"
9995 VERIFY "Llist": VERIFY "Llist"
```

This article was retyped with modifications for the Timex-Sinclair 2068 by:
David Solly,
Ottawa TSUG,
January 9, 1987.

Copal Screen Copier

By
David Solly

This screen copy program is a modification of a program written by John Bennett. The original program appeared in ZX Computing Monthly for February, 1987, pages 124 to 125. I have reworked the program to work on a Copal Write Hand L1200 in conjunction with the latest version of the Larken DOS.

How the program works

Line 1050 is a command to the Larken DOS to open stream number three to the line printer.

Line 1060 does a cold start on the Copal Write Hand.

Line 1070 de-selects the line-feed otherwise there would be gaps in the copy we are trying to make.

Line 1160 selects the 8/72" inch spacing mode on the Copal Write Hand.

Line 1180 puts the Copal Write Hand into graphics mode.

Line 1200 is a test to see if the Copal Write Hand is ready to print the next character.

Line 1210 is the actual work horse of the program. It uses the Out command to feed character codes directly to the Copal Write hand. If that were not done the printer would print Sinclair tokens for some of the graphics.

Lines 1250 to 1270 prints a few lines after the screen copy has been completed and returns the printer to the normal printing mode.

Program Listing
Timex-Sinclair 2068 and ZX Spectrum

```
1000 REM "OPUSCOPY: JOHN BENNETT
1010 REM "Modifications for Larken DOS and
Copal Write Hand L1200 printer by David
Solly, Ottawa, Ontario, Canada.
1020 BORDER 1: PAPER 6: CLS
1030 REM "COPAL COPY/Copal Write Hand L1200
1040 PRINT AT 5,0;"COPAL PRINTER COPY ROUTINE
1986": PRINT ''
1050 PRINT #4: OPEN #3,"lp": REM Changed for
Larken DOS.
1060 LPRINT CHR$ 27;CHR$ 64;: REM Cold start
printer.
1070 LPRINT CHR$ 19;: REM Deslects line-feed
1080 PRINT                       "Set your
printer up first then  LOAD your picture or
SCREEN$.   The printer will start up
      automatically after 15 seconds. It will
then take about six      minutes to print a
full screen"
1090 INPUT "Enter title of SCREEN to LOAD
";N$
1100 PRINT #1;AT 1,0;"   Load from Tape or
Disc? (T/D) ": PAUSE 0
1110 IF INKEY$="d" OR INKEY$="D" THEN GO TO
1140
1120 IF INKEY$<>"t" AND INKEY$<>"T" AND
INKEY$<>"d" AND INKEY$<>"D" THEN GO TO 1100
1130 LOAD N$SCREEN$ : GO TO 1160
1140 PRINT #4: LOAD N$SCREEN$ : REM Larken
Disc system.
1150 REM Copy
1160 LPRINT CHR$ 27;"A";CHR$ 8;
1170 FOR H=175 TO 0 STEP -8
1180 LPRINT CHR$ 27;"K";CHR$ 0;CHR$ 1;
1190 FOR W=0 TO 255
1200 IF IN 127=253 THEN GO TO 1200: REM
Waites for the printer to be ready.
1210 OUT 127,(128*POINT (W,H)+64*POINT
(W,H-1)+32*POINT (W,H-2)+16*POINT
(W,H-3)+8*POINT (W,H-4)+4*POINT
(W,H-5)+2*POINT (W,H-6)+POINT (W,H-7)): REM
Sends screen information directly to the
printer.
```

```
1220 NEXT W
1230 LPRINT CHR$ 10;
1240 NEXT H
1250 LPRINT : LPRINT : LPRINT
1260 LPRINT CHR$ 18;: REM Reset Line-Feed.
1270 LPRINT CHR$ 27;CHR$ 64;: REM Cold start
printer.
1280 REM Put here the codes for the format
the printer will need for the next task.
1290 STOP
1300 PRINT #4: SAVE "Copal.B1" LINE 1
```

10010